Final Project
Wei Wei

Introduction:

This dataset is “Hollywood's Most Profitable Stories”. (via. InformationIsBeautiful.net.) It includes the information of Title, genre, studio, profitability and ratings for movies released 2007-2012. I’m crazy about movie and this dataset is pretty interesting and informative to me. We can find the trendings of profit in the movie industry and which companies earn the most.
I wish to explore three most interesting questions:

1. Which company has the highest net profit and have the best investments.
2. What’s the trend of the movie profit in different marketplace and at different time.

Summary of Data:

Histogram:
From Figure 1, we can see the distributions of two score sources. The Rotten Tomatoes scores tend to be uniform distribution. While the audience scores approach to normal distribution.

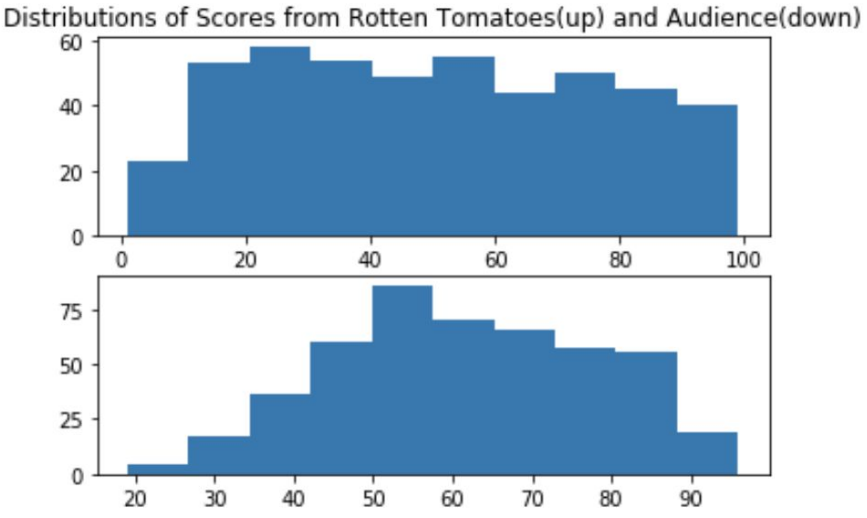


Figure 1. Distributions of Scores from Rotten Tomatoes and Audience

Barplot:
From figure 2, we can find that there are only 6 features having missing values and the Lead Studio has the most missing values. Since we have 600+ samples in total, we could safely drop these null values to get a more accurate understanding of the data.

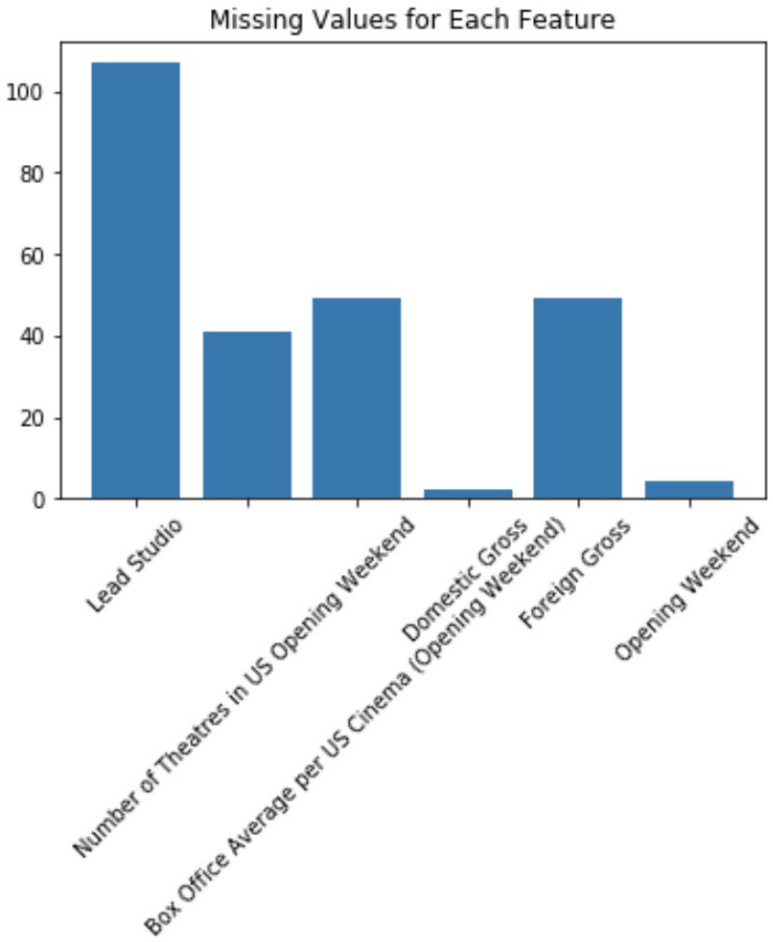


Figure 2. Missing Values for Each Feature

Boxplot

From Figure 3, we can find that budgets are generally lower than gross and gross has more large outliers.

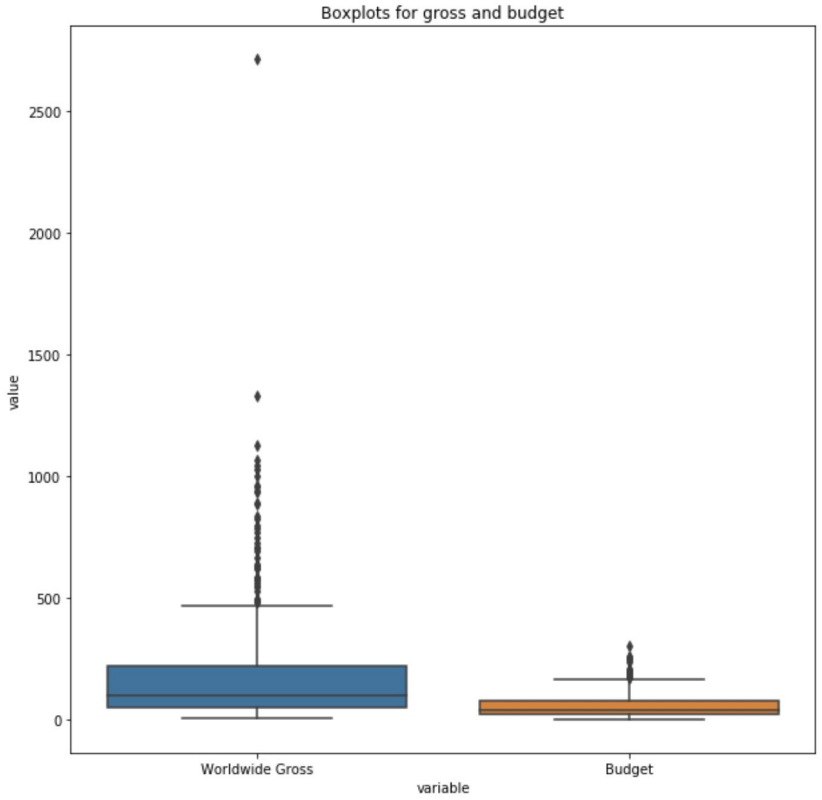


Figure 3. Boxplots for gross and budget

Scatterplot:
Generally, the budgets and worldwide gross have positive linear relationship, which makes sense.

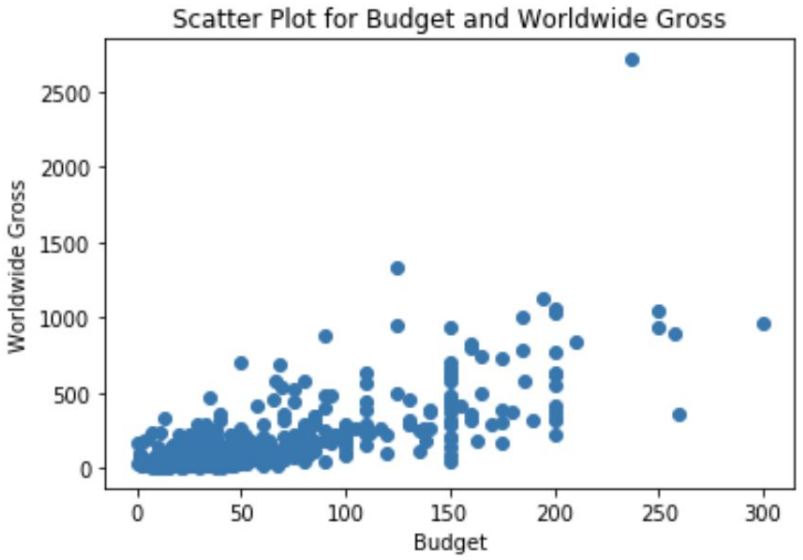


Figure 4. Scatter Plot for Budget and Worldwide Gross

Bubble Map
There is no obvious relationship between the budget/gross and the profitability. So, no matter what's the size of the movie, it can be both profitable or not.

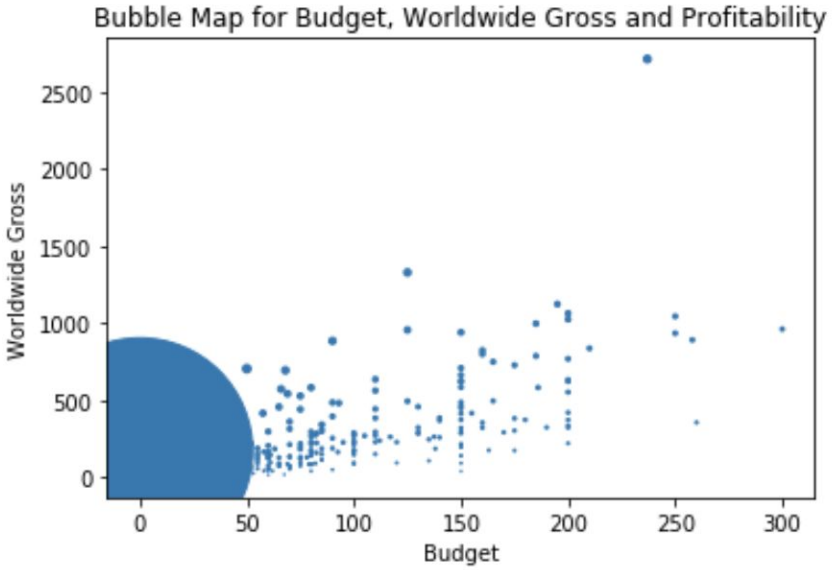


Figure 5. Bubble Map for Budget, Worldwide Gross and Profitability

Chloropleth Map

This map shows which states in US will give the most profit averagelly.

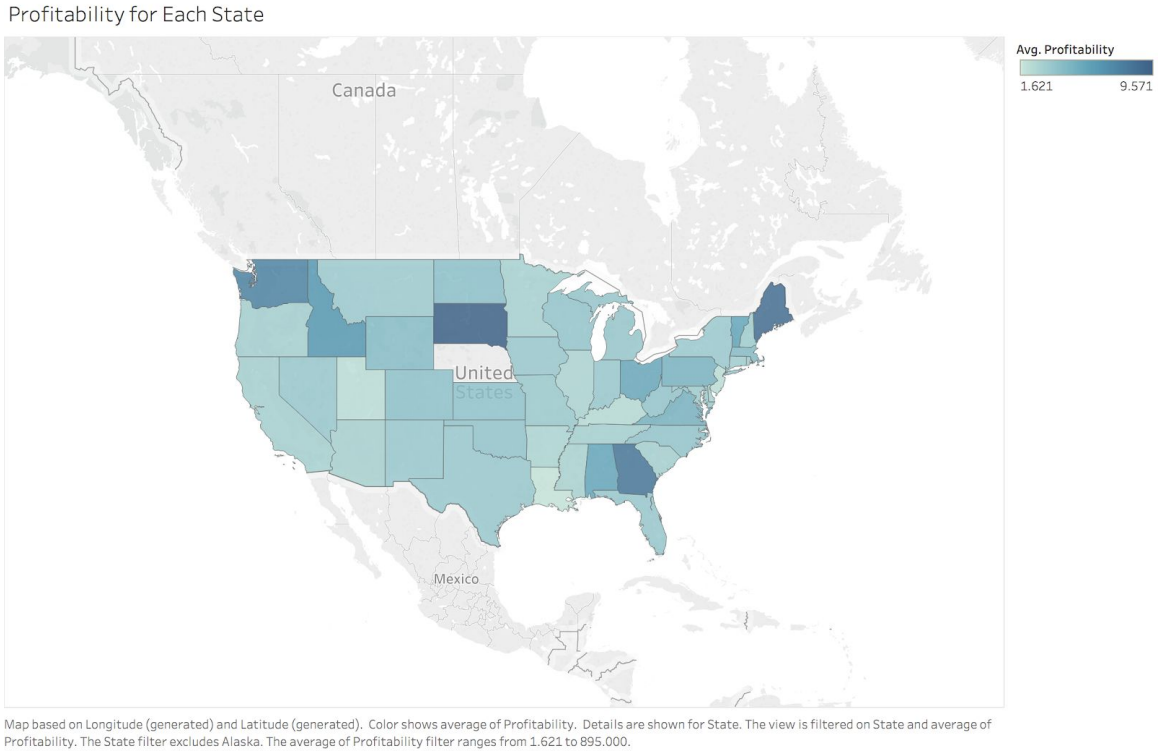


Figure 6. Chloropleth Map indicating Profitability for each State

Connection Map:

Connection map is not a good choice for my project. Cause there is no network with geographical data for the movies.

Heat map:

From Figure 7, we can see that Sony and Summit have the best profitability and the trends are increasing. While Fox and Disney have trends of decreasing.

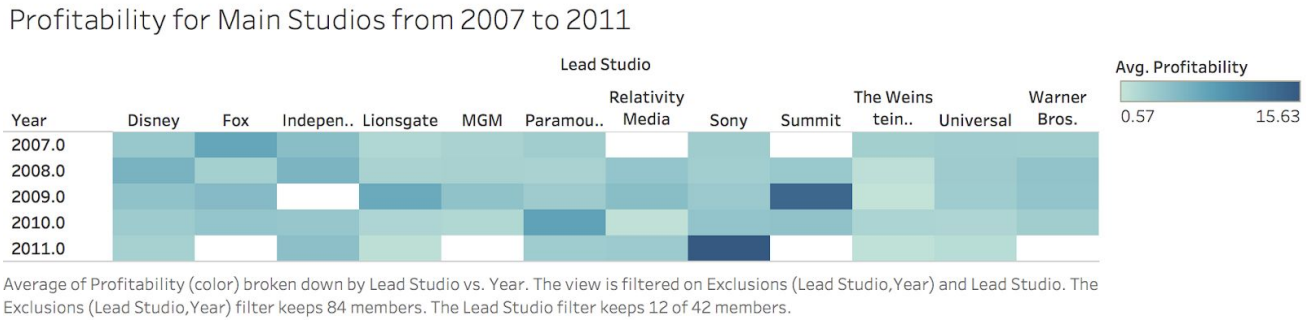


Figure 7. Profitability for Main Studios from 2007 to 2011

Stacked area or stream graph:

From Figure 8, we can find that the budgets for all the studios around year 2008 sharply decreased. It may be caused by that financial crisis. After 2008, only Disney and Universal Studio increased their budget significantly.

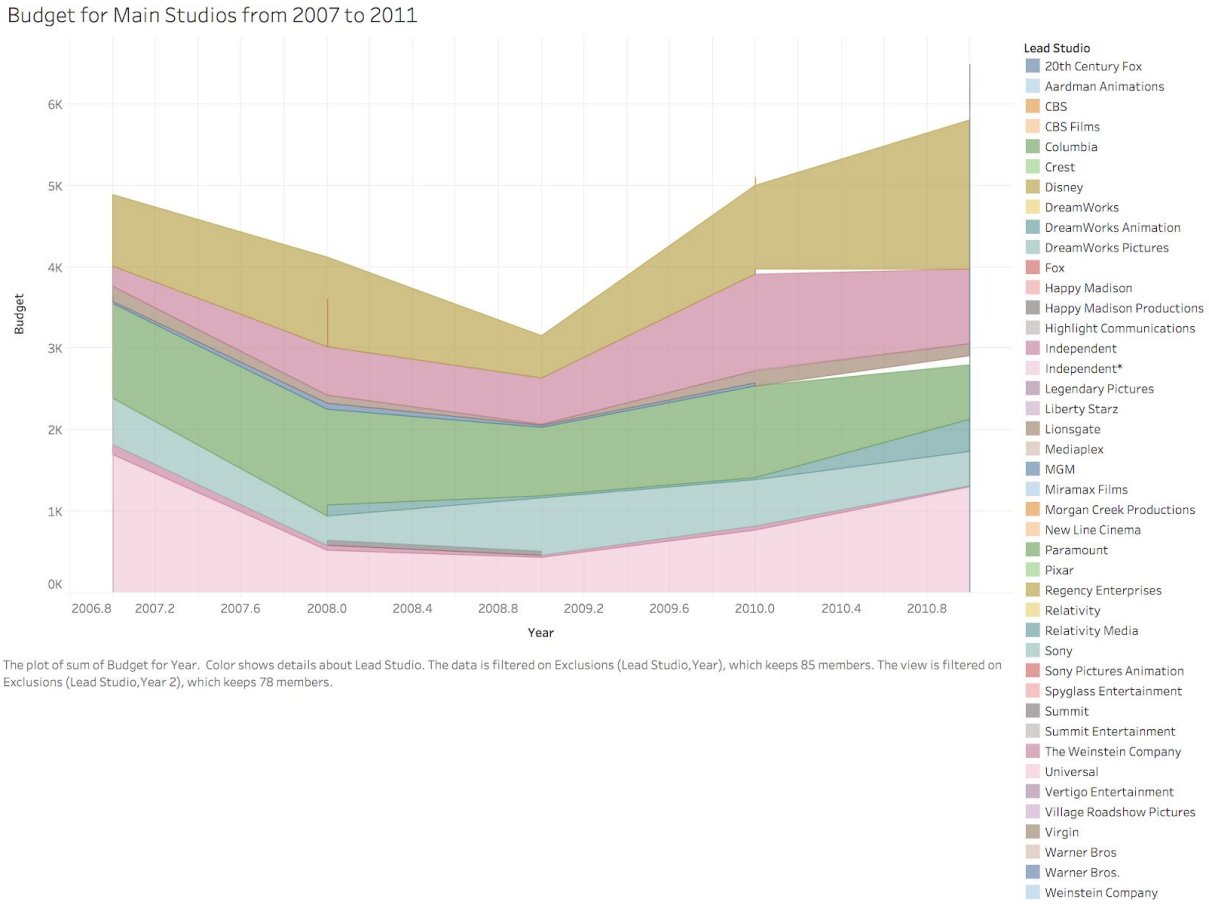


Figure 8. Budget for Main Studios from 2007 to 2011

Treemapping

From Figure 9, we can know the distribution of foreign profitability for different genres of movies.

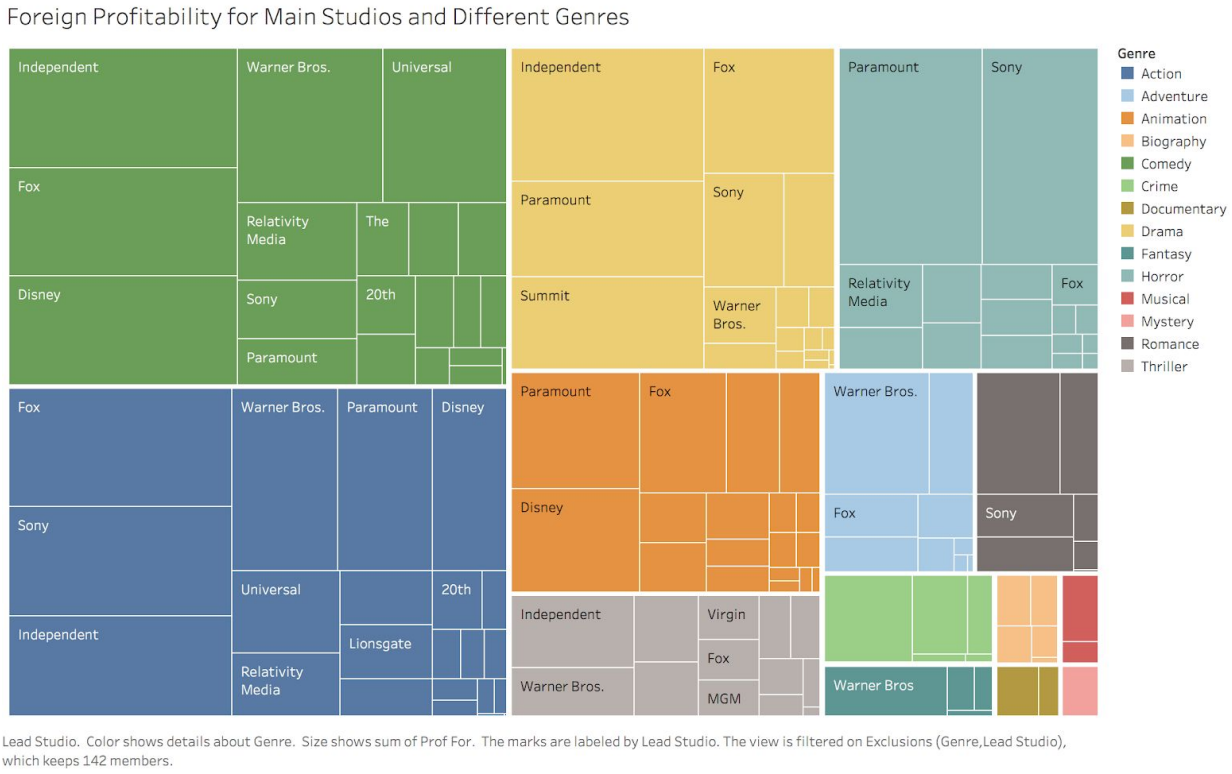
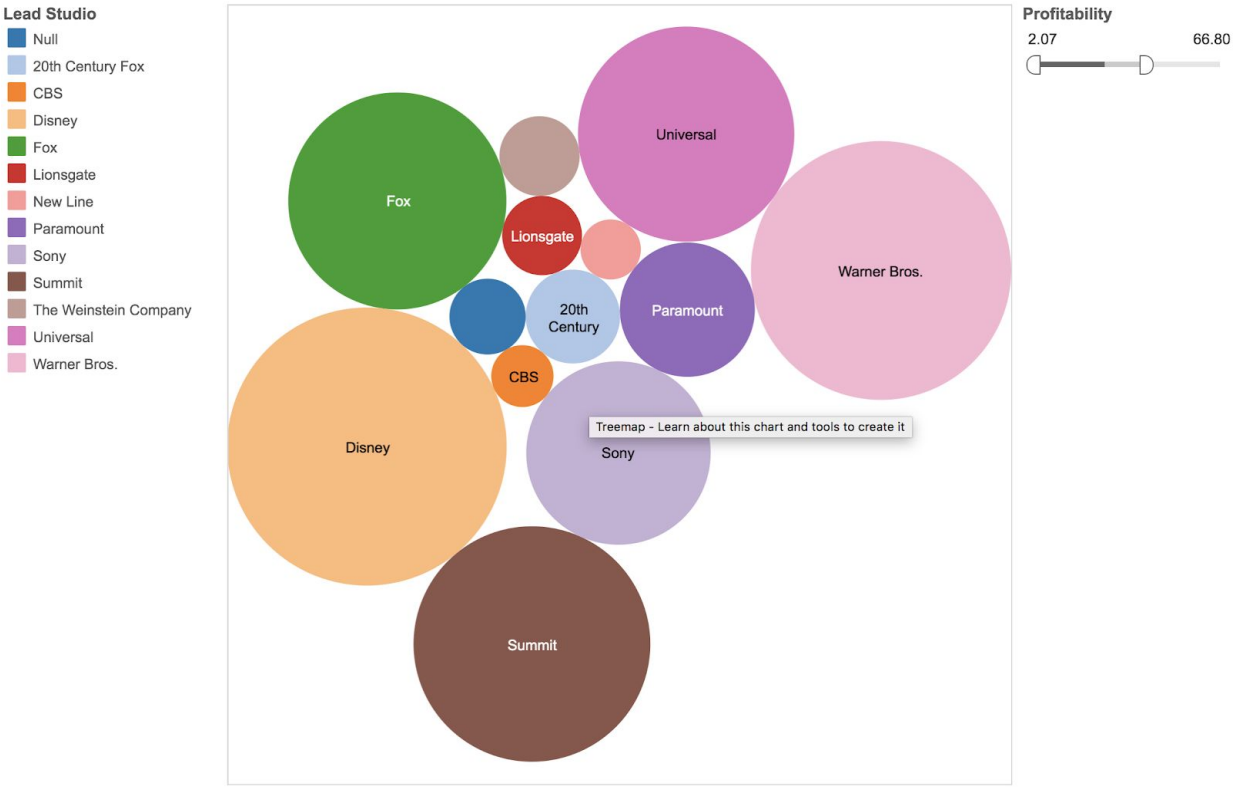


Figure 9. Foreign Profitability for Main Studios and Different Genres

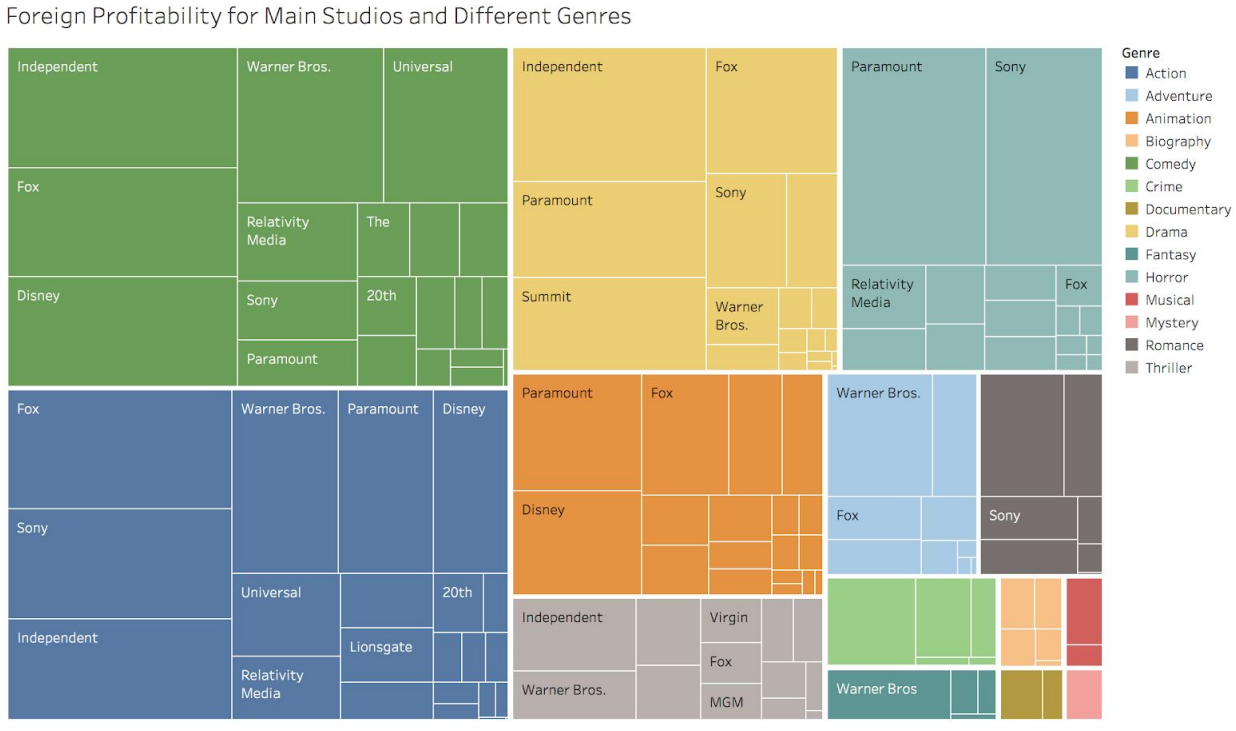
An interactive plot:
Same as Figure 9.

Storyline

I want to have a thourough understanding of the profitability of each studio. First, we want to have a look at how profitable are the studios generally with the bubble plot. We can find that Disney, Warner Bros, Universal, Summit and Fox are the 5 most profitable companies.

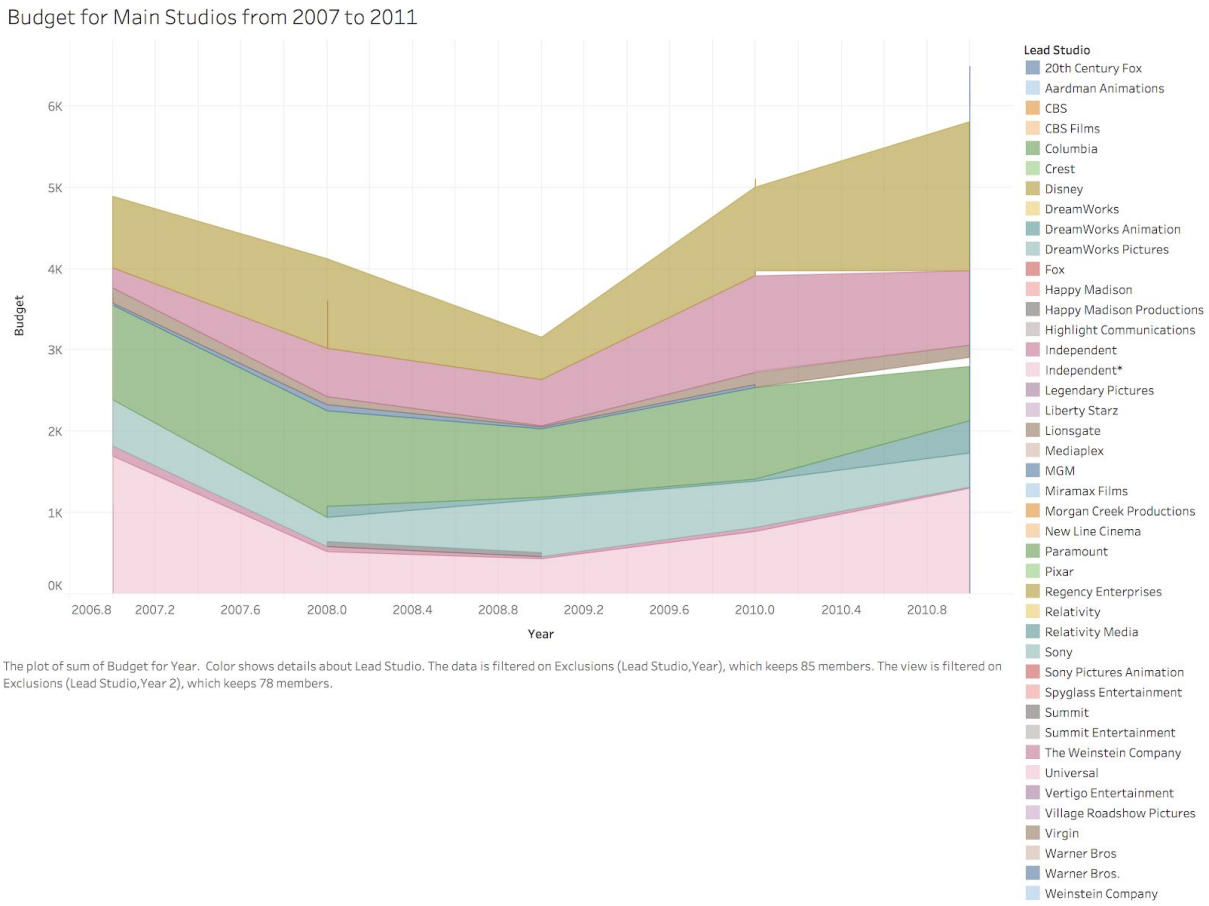


Surprisingly, though Disney has the largest overall profitability, the Foreign Profitability of Disney is behind a lot of studios in almost all the genres of movies.



Lead Studio. Color shows details about Genre. Size shows sum of Prof For. The marks are labeled by Lead Studio. The view is filtered on Exclusions (Genre,Lead Studio), which keeps 142 members.

From the stacked areas plot of investment. We can find that Disney, Universal, Sony, Fox have much higher investments than other studios. Recall that these studios are also the ones have highest profit rates. So, there is some kind of positive relationship between the scale of investment and the profitability in percentage.



The plot of sum of Budget for Year. Color shows details about Lead Studio. The data is filtered on Exclusions (Lead Studio,Year), which keeps 85 members. The view is filtered on Exclusions (Lead Studio,Year 2), which keeps 78 members.

Results/Summary/Conclusion

From the above visualization plots and analysis, we now have comprehensive understanding of the profitability of each studio at different time, for different genres and in different markets.

Appendix Containing All Code

Python codes are included below. Some other plots are done by Tableau. So no code is available.

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

executed in 1.88s, finished 11:19:52 2019-05-13

In [2]:

```
data = pd.read_csv("./Most Profitable Hollywood Stories.csv")
# data = data.dropna()
```

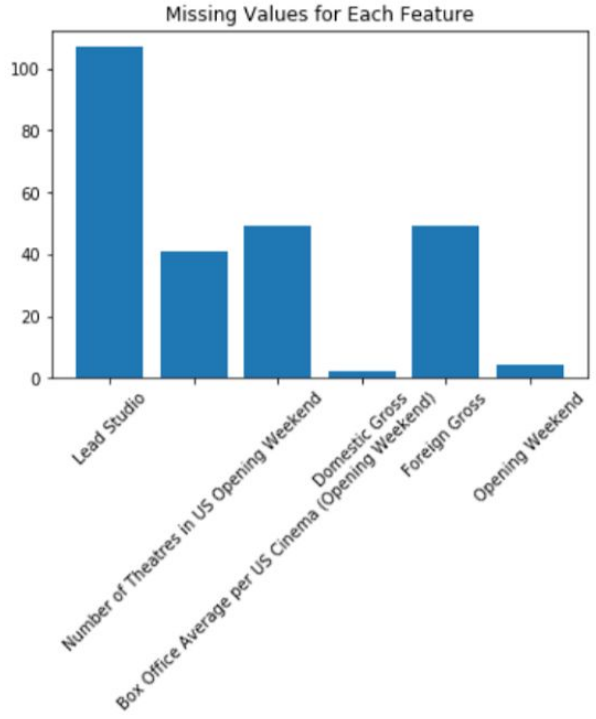
executed in 13ms, finished 11:19:52 2019-05-13

In [3]:

```
df_na = data.isnull().sum()
df_na = df_na[df_na>1]

x = list(range(6))
plt.title("Missing Values for Each Feature")
plt.bar(x, df_na)
plt.xticks(x, list(df_na.keys()), rotation=45)
plt.show()
```

executed in 251ms, finished 11:19:54 2019-05-13



In [4]:

```
data = data.dropna()
```

executed in 11ms, finished 11:19:55 2019-05-13

In [5]:

```
def s2f(x):
    try:
        return float(x)
    except:
        return None

data["Foreign Gross"] = data["Foreign Gross"].apply(s2f)
```

executed in 40ms, finished 11:20:01 2019-05-13

In [6]:

```
data = data.dropna()
```

executed in 6ms, finished 11:20:02 2019-05-13

In [7]:

```
data["prof_dome"] = data["Domestic Gross"]/data["Budget"]
data["prof_for"] = data["Foreign Gross"]/data["Budget"]
```

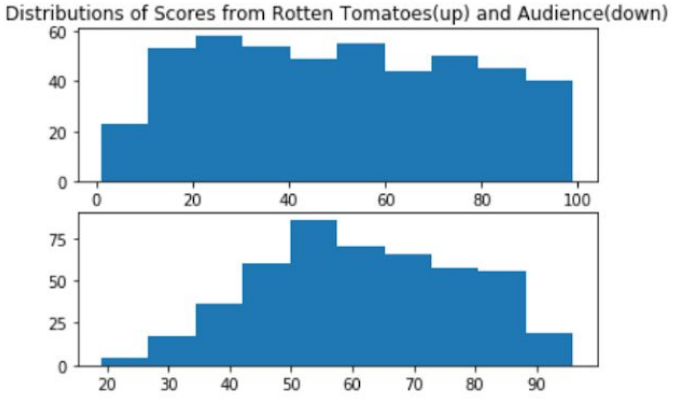
executed in 45ms, finished 11:20:03 2019-05-13

In [64]:

```
plt.subplot(2,1,1)
plt.title("Distributions of Scores from Rotten Tomatoes(up) and Audience(down)")
plt.hist(data["Rotten Tomatoes"])
plt.subplot(2,1,2)

plt.hist(data["Audience Score"])
# plt.title("Distributions of Scores from Audience")
plt.show()
```

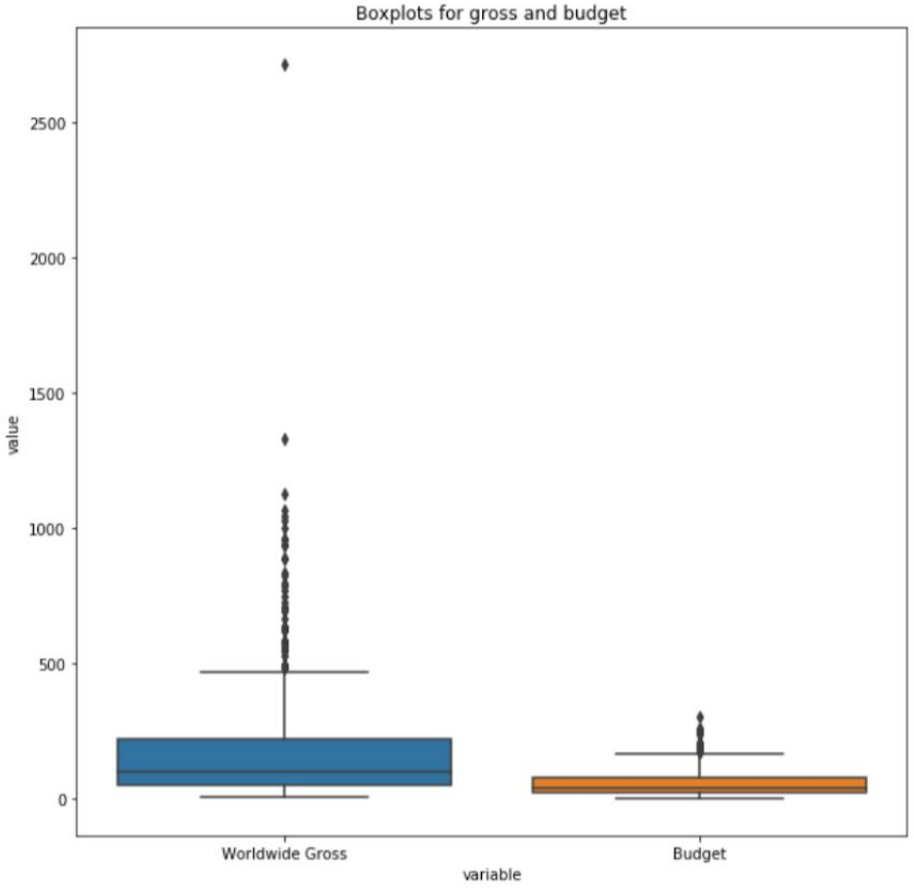
executed in 197ms, finished 20:55:26 2019-05-12



In [73]:

```
plt.figure(figsize=(10,10))
df_boxplot = data[["Worldwide Gross", "Budget"]]
sns.boxplot(x="variable", y="value", data=pd.melt(df_boxplot))
plt.title("Boxplots for gross and budget")
plt.show()
```

executed in 245ms, finished 21:07:33 2019-05-12



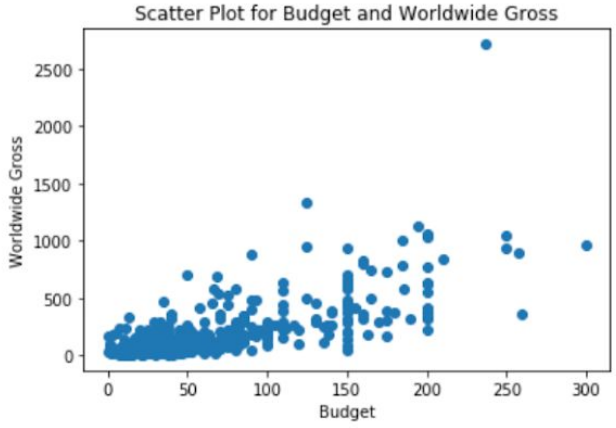
In [75]:

```
plt.scatter(data["Budget"], data["Worldwide Gross"])
plt.xlabel("Budget")
plt.ylabel("Worldwide Gross")
plt.title("Scatter Plot for Budget and Worldwide Gross")
```

executed in 178ms, finished 21:11:44 2019-05-12

Out[75]:

```
Text(0.5,1,'Scatter Plot for Budget and Worldwide Gross')
```



In [80]:

```
data["Profitability"] = data["Profitability"].apply(lambda x: float(x[:-1])/100)
```

executed in 77ms, finished 21:20:02 2019-05-12

```
/anaconda3/envs/ml/lib/python3.6/site-packages/ipykernel_launcher.py:
1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy (http://pandas.pyd
ata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy)
  """Entry point for launching an IPython kernel.
```

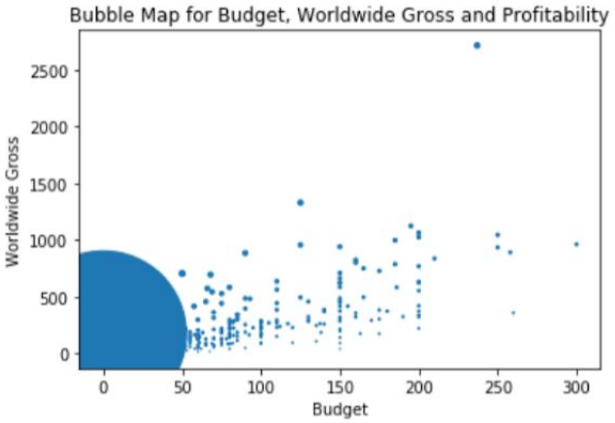
In [86]:

```
plt.scatter(data["Budget"], data["Worldwide Gross"], s=data["Profitability"])
plt.xlabel("Budget")
plt.ylabel("Worldwide Gross")
plt.title("Bubble Map for Budget, Worldwide Gross and Profitability")
```

executed in 193ms, finished 21:23:23 2019-05-12

Out[86]:

```
Text(0.5,1,'Bubble Map for Budget, Worldwide Gross and Profitability')
```



In [111]:

```
data.to_csv("finaldata.csv")
```

executed in 16ms, finished 22:35:20 2019-05-12

Link to your github page with this analysis:

https://github.com/weiweisf/Data-Visualization/blob/master/Data-Visualization-Final-Project-Wei_Wei.pdf

Citations:

- Data Source: https://github.com/gchan/hollywood-budgets/blob/master/public/data/Most%20Profitable%20Hollywood%20Stories.csv
- Treemap: https://datavizcatalogue.com/methods/treemap.html
- Heatmap via seaborn: https://seaborn.pydata.org/generated/seaborn.heatmap.html
- Bubble Plot: https://python-graph-gallery.com/bubble-plot/